

CLAIMS

1. In a Java computing environment, a method of customizing a Java
5 runtime environment for a Java application suitable for execution by a
virtual machine, said method comprising:
marking one or more Java Bytecodes associated with a Java class
file;
generating at least one attribute for said one or more marked Java
10 Bytecodes; and
loading at least one feature of Java runtime into said virtual machine
based on said at least one attribute.
2. A method as recited in claim 1, wherein said loading operates to load a
15 feature only if said feature has an associated attribute.
3. A method as recited in claim 1, wherein said marking is performed by a
Java compiler extension.
- 20 4. A method as recited in claim 1, wherein said marking is performed by a
software tool suitable for analyzing runtime performance of said Java
application in the runtime environment.
5. A method as recited in claim 1, wherein said one or more marked Java
25 Bytecodes are associated with a Java method.
6. A method as recited in claim 5, wherein said one or more marked Java
Bytecodes is associated with a Java object that has an attribute that is of
interest.
- 30 7. A method as recited in claim 6, wherein said attribute is the life span,
size, or class of said Java object.

8. A method as recited in claim 7, wherein said generating of said at least one attribute is performed by a software module that operates to generate said at least one attribute in an attributes table of said class file as the last attribute.

9. A method as recited in claim 1, wherein said method further comprises:
reading said generated at least one attribute in said class file.

10. A method as recited in claim 9, wherein said reading is performed by a software module which also operates to load said at least one feature of Java runtime into a virtual machine.

11. A Java computing environment suitable for execution of a Java application in a Java virtual machine, said Java computing environment comprising:

an first software module suitable for marking one or more Java Bytecodes associated with a Java class file;

a second software module suitable for generating at least one attribute for said one or more marked Java Bytecodes; and

a third software module suitable for loading at least one feature of Java runtime into said virtual machine based on said at least one attribute.

12. A Java computing environment as recited in claim 11, wherein said Java computing environment further comprises:

a fourth software module which can interact with the first, second, and third software modules.

13. A Java computing environment as recited in claim 12, wherein the fourth software module operates as a runtime performance manager and operates to ensures that said at least one feature is appropriately loaded into said virtual machine.

14. A Java computing environment as recited in claim 13, wherein said runtime performance manager includes a database that can be used as input by the second software module to generate said at least one attribute.
15. A Java computing environment as recited in claim 11, wherein said first software module is a compiler extension or a software tool suitable for analyzing a Java application.
16. A computer readable media including computer program code for customizing a Java runtime environment for a Java application suitable for execution in a virtual machine, said computer readable media comprising:
- computer program code for marking one or more Java Bytecodes associated with a Java class file;
 - computer program code for generating at least one attribute for said one or more marked Java Bytecodes; and
 - computer program code for loading at least one feature of Java runtime into said virtual machine based on said at least one attribute.
17. A computer readable media as recited in claim 16, wherein said computer program code for loading operates to load a feature only if said feature has an associated attribute.
18. A computer readable media as recited in claim 16, wherein said computer program code for marking is performed by a Java compiler extension.
19. A computer readable media as recited in claim 16, wherein said computer program code for marking is performed by a software tool suitable for analyzing performance of a Java application in the runtime environment.

20. A method as recited in claim 19, wherein said marked Java Bytecode is associated with a Java object that has an attribute that is of interest.

21. A method as recited in claim 20, wherein said attribute is the life span,
5 size, or class of said Java object.

SECRET